④

A Technical Report

FULL ANALYSIS OF A POWERFUL WINDOW RANDOM ACCESS ALGORITHM

Submitted to:

Office of Naval Research
Department of the Navy
800 N. Quincy Street
Arlington, VA   22217-5000

Submitted by:

P. Papantoni-Kazakos
Professor

M. Paterakis
Graduate Assistant

DTIC
S ELECTE D
MAR 1 5 1988
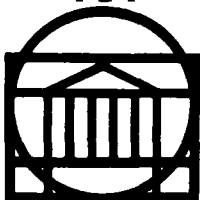
SCHOOL OF ENGINEERING AND

APPLIED SCIENCE

DEPARTMENT OF ELECTRICAL ENGINEERING

UNIVERSITY OF VIRGINIA

CHARLOTTESVILLE, VIRGINIA 22901

88 3 14 136

A Technical Report

FULL ANALYSIS OF A POWERFUL WINDOW
RANDOM ACCESS ALGORITHM

Submitted to:

Submitted by:

P. Papantoni-Kazakos
Professor

M. Paterakis
Graduate Assistant

Department of Electrical Engineering

SCHOOL OF ENGINEERING AND APPLIED SCIENCE

UNIVERSITY OF VIRGINIA

CHARLOTTESVILLE, VIRGINIA

Copy No.____

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION	1b. RESTRICTIVE MARKINGS
Unclassified	None

2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT
	Approved for Public Release;
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	Distribution Unlimited

4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
UVA/525415/EE88/112	

6a. NAME OF PERFORMING ORGANIZATION	6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION
University of Virginia Dept. of Electrical Engineering		Office of Naval Research Resident Representative

6c. ADDRESS (City, State and ZIP Code)	7b. ADDRESS (City, State and ZIP Code)
Thornton Hall Charlottesville, VA 22901	818 Connecticut Avenue, N. W. Eighth Floor Washington, DC 20006

8a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
Dept. of the Navy Office of Naval Research		N00014-86-K-0742

8c. ADDRESS (City, State and ZIP Code)	10. SOURCE OF FUNDING NOS.			
800 N. Quincy Street Arlington, VA 22217-5000	PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT NO.

11. TITLE (Include Security Classification)
see 16a

12. PERSONAL AUTHOR(S)
P. P. Kazakos/M. Paterakis

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Yr., Mo., Day)	15. PAGE COUNT
Technical	FROM 9/1/86 TO 8/31/88	1988 March	29

16. SUPPLEMENTARY NOTATION
Full Analysis of a Powerful Window Random Access Algorithm

17.	COSATI CODES		18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

In this paper, a simple full sensing window random access algorithm is analyzed in the presence of the limit Poisson user model. The throughput of the algorithm is 0.43, and its delay and resistance to channel errors characteristics are superior to those induced by the Capetanakis window algorithm. In addition, the simple operations of the algorithm, in conjunction with its regenerative properties, allow for the computation and evaluation of the output traffic interdeparture distribution. The latter is needed in the evaluation of interacting systems which use the algorithm for their internal transmissions.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	Unclassified

22a. NAME OF RESPONSIBLE INDIVIDUAL	22b. TELEPHONE NUMBER (Include Area Code)	22c. OFFICE SYMBOL
Dr. R.N. Madan		

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

TABLE OF CONTENTS

I. Introduction

In systems where independent users transmit through a single common channel, the deployment of random access transmission algorithms is frequently desirable, for the following reasons: (1) They are implemented independently by each user, without a priori coordination among the users. (2) They are insensitive to changing user population. (3) They induce low delays for low input rates, when the user traffic is bursty.

In this paper, we propose and analyze a full feedback sensing window random access algorithm. The algorithm was first proposed for systems with strict delay limitations, [3], and it requires that each user know the overall feedback history, (full feedback sensing). As compared to other such existing algorithms, the proposed algorithm has the following interesting properties: (1) It can be easily modified to operate under limited feedback sensing, where each user follows the feedback history from the time he generates a message to the time when this message is successfully transmitted. (2) When the Poisson user model is adopted, the algorithm attains the same throughput as that attained by the Capetanakis's dynamic algorithm, [1], while it induces lower delays for arrival rates above 0.30, and better resistance to feedback channel errors. (3) The simple operations of the algorithm allow analytical evaluation when strict delay limitations exist, [3]. Its simplicity, together with its regenerative properties, provide the means for the analytical evaluation of the output traffic interdeparture distribution induced by the algorithm. The analysis of the latter distribution is important when several systems which use some Random Access Algorithm, (RAA), for internal transmissions interact, and it is not quite feasible when either the Capetanakis's, [1], or the Gallager's, [2], algorithms are deployed. (4) As compared to Gallager's algorithm, [2], the proposed algorithm operates in systems where the Poisson user model is not valid, (e.g., when more than one packets can be generated within a given time instant), and can be then analyzed.

The organization of the paper is as follows: In section II, the system model is presented, and the algorithm is described. In section III, throughput and delay analysis for the Poisson user model is included. In section IV, the performance of the algorithm in the presence of feedback channel errors and its operation under limited feedback sensing are discussed. In section V, the output traffic interdeparture distribution induced by the algorithm is analytically evaluated. In section VI, some conclusions are drawn.

II. The System Model and the Algorithm

We assume packet transmitting users, slotted channel, binary collision versus noncollision, (CNC), feedback per slot, and zero propagation delays. and absence of feedback errors. We also assume that collided packets are fully destroyed and retransmission is then necessary. Time is measured in slot units, slot t occupies the time interval $[t, t+1)$, and x_t denotes the feedback that corresponds to slot t; $x_t = C$ and $x_t = NC$ represent then collision and noncollision slot t, respectively. For this system, let the following full feedback sensing synchronous random access algorithm be deployed.

The algorithm utilizes a window of length Δ. Let t be a time instant that corresponds to the beginning of a slot such that, for some $t_1 < t$, all the packet arrivals in $(0, t_1]$ have been successfully transmitted by the algorithm and there is no information regarding the arrival interval $(t_1, t]$. The instant t is then called Collision Resolution Point, (CRP), the arrival interval $(0, t_1]$ is called "resolved interval", and the interval $(t_1, t]$ is called "the lag at t". In slot t, the packet

arrivals in $(t_1, t_2 \overset{\Delta}{=} \min (t_1 + \Delta, t)]$ attempt transmission, and the arrival interval $(t_1, t_2]$ is then called the "examined interval". If $(t_1, t_2]$ contains at most one packet, then it is resolved at t. If $(t_1, t_2]$ contains at least two packets, instead, then $x_t = C$, a collision occurs at t, and its resolution starts with slot t+1. Until the collision at t is resolved, no packets that have arrived after t_2 are allowed transmission. The time period required for the resolution of the latter collision is called the Collision Resolution Interval, (CRI). If the examined interval contains at most one packet, then the CRI lasts one slot. During some CRI which starts with a collision slot, each user acts independently via the utilization of a counter whose value at time t is denoted r_t. When a user transmits for the first time he sets $r_t = 1$. The counter values can be either 1 or 2, and they are updated and utilized according to the rules below.

1. The user transmits in slot t, if and only if $r_t = 1$. A packet is successfully transmitted in t, if and only if $r_t = 1$ and $x_t = NC$.

2. The counter values transition in time as follows:

   (a) If $x_{t-1} = NC$ and $r_{t-1} = 2$, then $r_t = 1$

   (b) If $x_{t-1} = C$ and $r_{t-1} = 2$, then $r_t = 2$

   (c) If $x_{t-1} = C$ and $r_{t-1} = 1$, then

$$r_t = \begin{cases} 1, \ w.p. \ 0.5 \\ 2, \ w.p. \ 0.5 \end{cases}$$

From the above rules it can be seen that a CRI which starts with a collision slot ends with two consecutive noncollision (NC) slots. Furthermore, two consecutive NC slots can not occur at any other instant during a CRI. Thus, the end of a such a CRI can be identified by all the users in the system, upon the observation of two consecutive NC slots.

Remarks We note that the algorithmic operations can be depicted by a two-cell stack, where at each time instant t, cell 1 contains the transmitting users, (those with $r_t = 1$), and cell 2 contains the withholding users, (those with $r_t = 2$). The algorithm lumps, thus, the unsuccessful users together. In contrast, Capetanakis's algorithm distributes the unsuccessful users across the cells of an infinite-cell stack. As with Capetanakis's dynamic algorithm, the window size Δ is here subject to optimization for throughput maximization.

III. Throughput and Delay Analysis

In this section, we present throughput and delay analysis of the algorithm. We assume continuous feedback sensing and we adopt the Poisson user model. Indeed as proven in [5], for a large class of random access algorithms (RAAs), as the population size increases the stability of an algorithm in the class is determined by its throughput under the Poisson user model.

Consider the system model and the algorithm in Section II. Let the system start operating at time zero. Let t_i ; $i \geq 1$ be the sequence of successive CRPs, and let X_i be the lag at t_i. The

sequence X_i ; $i \geq 1$ is a Markov Chain with state space $\mathbf{F}$, $\mathbf{F}$ is an at most denumerable subset of the interval $[1, \infty)$. It can be seen that any state can be reached from any other; therefore X_i ; $i \geq 1$ is an irreducible Markov Chain. Since $P(X_{i+1} = 1 \mid X_i = 1) > 0$, we conclude that X_i; $i \geq 1$ is also aperiodic. Thus Pake's Lemma [11] applies, and gives that the following condition is sufficient for the ergodicity of the Markov Chain, (stability of the system):

$$E(l \mid \Delta, d) < \Delta \qquad (1)$$

where $E(l \mid \Delta, d)$ denotes the expected length of a CRI, given that it starts with an examined interval of length Δ and with a lag of length d.

Since the Markov Chain is uniformly downward bounded (i.e. there exists a constant m such that the transition probabilities p_{kj} satisfy $p_{kj} = 0$ for $j < k - m$. Here $m = \Delta - 1$), Kaplan's Theorem [12] applies and gives that:
If

$$E(l \mid \Delta, d) > \Delta$$

then the Markov Chain is not ergodic and the system is unstable.

Let L_k denote the expected length of a CRI given that it starts with a collision of multiplicity k. We can then write:

$$E(l \mid \Delta, d) = \sum_{k=0}^{\infty} E(l \mid \Delta, d, k) \, e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!} = \sum_{k=0}^{\infty} L_k e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!} \qquad (2)$$

since

$$E(l \mid \Delta, d, k) = L_k \qquad (2a)$$

depends only on k.
In the Appendix we show that:
(i) L_k ; $k \geq 0$ can be computed recursively, and
(ii) L_k are quadratically upper bounded, $L_k \leq L_k^u = \alpha k^2 + \beta k + \gamma$; $k \geq 2$.
Expression (1) together with (i) and (ii) are used in the computation of the algorithmic throughput.

We define the delay D_n, experienced by the n-th packet as the time difference between its arrival and the end of its successful transmission. We are interested in evaluating the first moment of the steady state delay process, when it exists. Let $T_1 = 1$, $X_1 = 1$, and define T_{i+1} as the first CRP after T_i at which the lag has length one. From the description of the algorithm it can be seen that the induced delay process probabilistically restarts itself at the beginning of each slot T_i, $i = 1, 2, \ldots$ The interval $[T_i, T_{i+1})$ will be referred to as the i-th session. Note that the sessions have lengths that are i.i.d random variables.

Let R_i ; $i = 1, 2, \ldots$, denote the number of packets successfully transmitted in the interval $(0, T_i]$; (note that R_i also represents the number of packets arrived during the interval $[0, T_i - 1)$, since T_i is a CRP at which the lag has length one). Then, $Q_i = R_{i+1} - R_i$; $i \geq 1$, is the number of packets successfully transmitted in the interval $(T_i, T_{i+1}]$, these are the packets that arrived during the interval $[T_i - 1, T_{i+1} - 1)$. The sequence R_i ; $i \geq 1$, is a renewal process since Q_i ; $i \geq 1$, is a sequence of nonnegative i.i.d random variables. Furthermore, the delay process D_n ; $n \geq 1$, is regenerative with respect to the renewal process R_i ; $i \geq 1$, with regeneration cycle Q_1. From the regenerative theorem [4], we conclude that if $Q = E(Q_1) < \infty$ and $W = E \left\{ \sum_{i=1}^{Q_1} D_i \right\} < \infty$, then there

exists a real number **D** such that,

$$\mathbf{D} = \lim_{n \to \infty} n^{-1} \sum_{i=1}^{n} D_i = \lim_{n \to \infty} n^{-1} E\left(\sum_{i=1}^{n} D_i\right) = \frac{W}{Q} \quad w.p. \ 1 \tag{3}$$

In addition, since $P(Q_1 = 1) > 0$, the distribution of C_1 is aperiodic and there exists a random variable D_∞ such that the sequence D_n ; $n \geq 1$ converges in distribution to D_∞. D_∞ represents the steady state delay induced by the algorithm and its mean satisfies the equality

$$E(D_\infty) = \frac{W}{Q} \tag{4}$$

The quantity **D** will be referred to as the mean packet delay. From (4) we observe that the mean packet delay can be determined by computing the quantities of the right hand side of the equality. In the Appendix we develop two systems of linear equations whose solution may be used to compute the mean cycle length Q and the mean cumulative delay W.

In Table 1, we include the computed upper and lower bounds, $\mathbf{D}^u$ and $\mathbf{D}^l$ respectively, on the mean packet delay **D**, for various Poisson intensities λ, and for both the proposed and the Capetanakis's dynamic algorithms.

	Proposed algorithm		Capetanakis dynamic algorithm	
λ	D^l	D^u	D^l	D^u
0.02	1.562	1.563	1.563	1.564
0.06	1.708	1.716	1.713	1.719
0.10	1.888	1.917	1.903	1.921
0.16	2.257	2.363	2.308	2.362
0.20	2.607	2.812	2.712	2.809
0.24	3.103	3.467	3.308	3.476
0.30	4.412	5.197	4.976	5.365
0.32	5.162	6.170	5.973	6.501
0.36	7.941	9.665	9.798	10.883
0.38	11.008	13.398	14.121	15.855
0.40	18.262	22.024	24.427	27.736
0.42	57.354	67.665	78.530	90.212

Table 1

Upper and Lower Bounds on Steady-State Expected Delays

Regarding the throughput λ^* and the optimal window size Δ^*, the following results were found.

Proposed Algorithm:	$\lambda^* = 0.4295$	$\Delta^* = 2.33$
Capetanakis' Dynamic Algorithm:	$\lambda^* = 0.4295$	$\Delta^* = 2.677$

Table 2

Throughput and Optimal Window Sizes

From Table 2, we observe that, for the Poisson user model, the algorithm in this paper attains the same throughput as the Capetanakis's dynamic algorithm, but utilizes a smaller window size. From Table 1, we observe that the two algorithms induce practically identical delays for Poisson rates in (0, 0.30), while for Poisson rates in (0.30, 0.42], the proposed algorithm induces lower delays.

Remarks It may seem surprising that the algorithm in this paper attains the same throughput as the Capetanakis's dynamic algorithm, and that it outperforms the latter in terms of delay performance. Indeed, the expected lengths L_k in (2a) are bounded by quadratic expressions, while the same lengths, L'_k, for the Capetanakis algorithm are bounded by linear functions of k. However, $L_2 = 4.5$ while $L'_2 = 5$. At the same time, since $\Delta^* = 2.33$ for the proposed algorithm, the probability of a higher than two multiplicity collision is very small. The multiplicity-two events thus prevail, and the algorithm in this paper becomes better than the Capetanakis's algorithm. We note that, as found in [3], the algorithm performs very well in environments where strict delay limitations exist. Then, it allows significant improvement in delay performance, at the expense of minimal traffic loss. In addition, the analysis of the algorithmic performance when strict delay limitations exist is relatively simple, while the same analysis for the algorithms in [1] and [2] is still an open problem.

IV. Performance in the Presence of Feedback Errors and Operations in the Limited Sensing Environment

In this section, we study two important characteristics of the algorithm. Namely, its performance in the presence of feedback channel errors, and its operation and performance under limited feedback sensing.

Performance in the Presence of Feedback Errors

Let us assume that due to noisy conditions, the following types of feedback errors may occur: With probability ε an empty slot may be seen by the users as a collision slot. Also, with probability δ a slot occupied by a single transmission may be seen by the users as a collision slot. Let us also assume that a collision slot is always recognized correctly by the users. We consider the case where the probabilities ε and δ, being system characteristics, are known a priori. Then given ε and δ, the window size Δ is optimized for throughput maximization. We performed throughput analysis, (the details are included in the Appendix), for both the proposed algorithm

and the Capetanakis's dynamic algorithm, [1]. We exhibit the results in Table 3. From Table 3, we conclude that the proposed algorithm is very insensitive to feedback errors. Even for the practically extreme case $\varepsilon = \delta = 0.1$, the throughput is almost 90% of its value in the error free case. We also conclude that the proposed algorithm allows operation (positive throughput) as long as $\varepsilon < 1$ and $\delta < 1$, while if $\varepsilon \geq 0.5$ the throughput for the Capetanakis's algorithm is then zero. We notice for example that for $\varepsilon = 0.5$ and $\delta = 0$, the proposed algorithm attains throughput as high as $\lambda^* = 0.325$.

ε	δ	λ^* proposed alg.	λ^* Cap.
0.00	0.00	0.4295	0.4295
0.00	0.01	0.4248	0.4258
0.00	0.10	0.3873	0.3920
0.00	0.20	0.3463	0.3535
0.00	0.40	0.2655	0.2731
0.00	0.50	0.2251	0.2310
0.01	0.00	0.4272	0.4272
0.10	0.00	0.4117	0.4043
0.20	0.00	0.3930	0.3706
0.40	0.00	0.3503	0.2329
0.45	0.00	0.3382	0.1524
0.50	0.00	0.3503	0.0000
0.60	0.00	0.3050	0.0000
0.70	0.00	0.2750	0.0000
0.80	0.00	0.2280	0.0000
0.90	0.00	0.1700	0.0000
0.10	0.10	0.3706	0.3672
0.20	0.20	0.3139	0.2972
0.30	0.30	0.2589	0.2166
0.40	0.40	0.2064	0.1205
0.30	0.50	0.1885	0.1511
0.30	0.70	0.1183	0.0886
0.90	0.90	0.0105	0.0000

Table 3

Throughputs as a function of ε and δ. Window sizes optimized for every pair (ε, δ)

In some systems, the probabilities ε and δ may not be known a priori. In this case, an algorithm may be designed subject to the assumption of error free feedback. We found the throughputs for the proposed and the Capetanakis's dynamic algorithms, in this case, for various values of the error probabilities; that is, the maximum Poisson rates for which (1) is satisfied subject to the constraint that the windows are those in Table 2. Our results are shown in Table 4.

ε	δ	λ^* prop. alg.	λ^* Cap.
0.00	0.00	0.429	0.429
0.00	0.01	0.424	0.410
0.00	0.10	0.386	0.391
0.00	0.40	0.255	0.262
0.01	0.00	0.427	0.427
0.10	0.00	0.408	0.401
0.20	0.00	0.377	0.355
0.3	0.00	0.320	0.248
0.3305	0.00	0.289	0.000
0.37	0.00	0.214	0.000
0.3787	0.00	0.000	0.000
0.10	0.10	0.363	0.361
0.10	0.20	0.318	0.318
0.10	0.30	0.273	0.272
0.20	0.10	0.328	0.311
0.20	0.20	0.279	0.266
0.20	0.30	0.232	0.221
0.30	0.10	0.261	0.192
0.30	0.20	0.205	0.138
0.30	0.30	0.153	0.088

Table 4

Throughputs as a function of ε and δ. Window sizes unchanged for every pair (ε, δ)

From Table 4, we observe that the proposed algorithm is better than the Capetanakis's algorithm. For example the maximum ε value for which the proposed algorithm is stable is 0.378, while the Capetanakis's algorithm becomes unstable for $\varepsilon \geq 0.33$.

Operations in the Limited Sensing Environment

In the limited sensing environment, it is required that each user monitor the channel feedback only from the time he generates a packet, to the time this packet is successfully transmitted. Therefore, the users' knowledge of the channel feedback history is then asynchronous. The objective in this case is to prevent new arrivals from interfering with some collision resolution in progress. This is possible, if each user can decide whether a collision resolution is in progress or not, within a finite number of slots from the time he generates a new packet. We observe that a user who has a new packet and observes a C slot decides to wait, since he can then deduce that there is some collision resolution in progress. Also, as we explained in Section II, the occurrence of two consecutive NC slots corresponds to either two consecutive unit-length CRIs or to the end of a CRI which started with a collision slot; thus, upon the observation of such an event, a user can decide that there is no collision resolution in progress. In view of the above, we

conclude that under limited feedback sensing the algorithm can be modified to operate as follows:

The window size is the same as in the full feedback sensing case. However, the window slides through the unexamined interval from present to past and its edge is maintained one slot before the current time, (see Figure 2). Within each window, the operations of the algorithm are the same as in the full feedback sensing case.

In the limited sensing environment, and for very light input traffic, the algorithm induces mean packet delay equal to 2.5. As the rate of the input traffic increases, the mean delays approach those induced under full feedback sensing. The throughput of the algorithm remains identical to that under full feedback sensing. In Figure 2, we plot the expected delays that the algorithm induces, under both full and limited feedback sensing. In the latter case, the mean delays were computed by using methodologies similar to those in [4], [9].

Remarks We point out that the modification of Capetanakis's dynamic algorithm, for operations under limited feedback sensing, is still an open and complex problem. In contrast, the same modification is simple when the proposed algorithm is adopted. In systems where the Poisson user model is valid, the Part-and-Try algorithm with binary feedback in [8], can be modified to operate under limited feedback sensing, [9], [10]. The throughput of the latter algorithm is then 0.45. But when the Poisson user model is not valid, it leads to deadlocks. In contrast, the proposed algorithm does not lead to deadlocks and its performance can be then analytically evaluated.

V. The Output Traffic Interdeparture Distribution

In this section, we concentrate on the computation of the output traffic interdeparture distribution. In particular, we find analytically the steady-state distribution of the distance between two consecutive, successful transmissions, when the algorithm in this paper is deployed. We point out that the algorithm generates an output traffic process with memory. Our computations correspond thus to the first order distribution from this process. This first order distribution, together with a memoryless assumption, can be used as an approximation of the actual output traffic process, when studies of systems which deploy the algorithm and interact with each other are undertaken. Such interactions may correspond, for example, to: (1) Servicing the output traffic from several systems that deploy the algorithm, by a single server queue. (2) Transmitting the output traffic from a system that deploys the algorithm, through the transmission channel of another random access system, (multi-hop problem).

Our methodology utilizes the regenerative character of the output traffic process that the algorithm generates, and its steps are as those in [4]. We define the sequence $\{P_i\}_{i \geq 1}$ as follows: Each P_i is a collision resolution point, (CRP), which follows a slot containing a successful transmission and at which the lag equals one. P_1 is the first after zero such CRP, and for every $i \geq 1$, P_{i+1} is the first after P_i such CRP. Let S_i, $i \geq 1$ denote the number of successful transmissions in $(0, P_i]$, and let d_n denote the distance between the (n-1)-th and the n-th successful transmission. Then, S_i, $i \geq 1$ is a renewal process, and the process d_n, $n \geq 1$ is regenerative with respect to it.

Let us define, $C_i = S_{i+1} - S_i$, $i \geq 1$. Then C_i denotes the number of successful transmissions in the interval $(P_i, P_{i+1}]$, where this interval will be called the i-th cycle. Let us define,

$$I_n(s) = \begin{cases} 1 & , \text{ if } d_n = s \\ 0 & , \text{ otherwise} \end{cases} \tag{5}$$

$$H = E\{P_{i+1} - P_i\} \tag{6}$$

From the regenerative theorem [4], we then conclude that if $C = E\{C_1\} < \infty$, then,

$$\lim_{N \to \infty} N^{-1} \sum_{n=1}^{N} I_n(s) = \lim_{N \to \infty} N^{-1} E\{\sum_{n=1}^{N} I_n(s)\} = C^{-1} E\{\sum_{n=1}^{C_1} I_n(s)\} \tag{7}$$

; where if the intensity of the input Poisson traffic is λ, then,

$$C = \lambda H \tag{8}$$

In addition, since $P(C_1 = 1) > 0$, the distribution of C_1 is aperiodic and there exists a random variable d_∞, such that the sequence d_n, $n = 1, 2, \ldots$ converges in distribution to d_∞. Then, d_∞ represents the steady state interdeparture distance induced by the algorithm, and its distribution satisfies the equality,

$$P(d_\infty = s) = C^{-1} E\{\sum_{n=1}^{C_1} I_n(s)\} \tag{9}$$

The finiteness and the computation of the quantities C and $E\{\sum_{n=1}^{C_1} I_n(s)\}$ in (9) are related to the existence and computation of appropriate solutions to infinite-dimensionality linear systems. Those systems and their solutions are included in the Appendix. In Table 5, we include the computed upper and lower bounds, respectively denoted P_s^u and P_s^l, on the probability $P(d_\infty = s)$, for input traffic Poisson intensities $\lambda = 0.1$ and $\lambda = 0.4$. In Figure 3, we plot the lower bounds against s, for various input traffic Poisson intensities λ.

s	$\lambda = 0.1$		$\lambda = 0.4$	
	P_s^l	P_s^u	P_s^l	P_s^u
1	0.1420	0.1427	0.4702	0.4728
2	0.0816	0.0832	0.2000	0.2048
3	0.0704	0.0739	0.0998	0.10806
4	0.0641	0.0696	0.0612	0.0787
5	0.0537	0.0603	0.0401	0.0603
6	0.0502	0.0591	0.0280	0.0397
7	0.0420	0.0503	0.0196	0.0264
8	0.0364	0.0452	0.0099	0.0173
9	0.0332	0.0431	0.0057	0.0094
10	0.0265	0.0393	0.0013	0.0062

<u>Table 5</u>

Upper and lower bounds on the interdeparture distribution for Poisson rates
$\lambda = 0.1$ and $\lambda = 0.4$

From the results in Table 5 and Figure 3, we draw the following conclusions:

(1) For low rates λ of the Poisson input traffic, ($\lambda \leq 0.1$), the interdeparture distribution is close to the Bernoulli distribution whose parameter is $p = \lambda e^{-\lambda}$. In particular, denoting $P_s \overset{\Delta}{=} P(d_\infty = s)$, we then have $P_s \approx p (1-p)^{s-1}$, for $s \geq 2$. The probability P_1, however, is then significantly larger than the Bernoulli parameter p. The intuitive explanation of the latter phenomenon goes as follows: For small rates λ, single arrivals in two consecutive slots occur with probability $p^2 = (\lambda e^{-\lambda})^2 \approx \lambda^2$, while the probability of a collision slot is then approximately equal to $2^{-1} \lambda^2 e^{-\lambda} \approx 2^{-1} \lambda^2$. Thus, for small rates λ, single arrivals in two consecutive slots contribute two thirds·of P_1, while the remaining one third is due to consecutive departures at the end of a collision resolution interval.

(2) As the rate λ of the Poisson input traffic increases, the interdeparture distribution induced by the algorithm deviates further from the Bernoulli distribution. In fact, as λ increases, the mass of the interdeparture distribution accumulates at relatively small s values. For example, for $\lambda = 0.4$, we have $P_1 = 0.471$ and $\sum_{s=1}^{10} P_s \approx 1$.

<u>Remarks</u> Our results show that it is generally wrong to conjecture Bernoulli interdeparture distribution. In fact, the obtained distribution is far from Bernoulli. Even for small input Poisson rates λ, the probability P_1 does not equal to $\lambda e^{-\lambda}$. We point out that the proposed method could be applied to other algorithms as well, including the Capetanakis's algorithm. However the development, of the corresponding recursions and infinite dimensionality linear systems of equations, is then very complicated. The simple operations of the proposed algorithm present an advantage, which does not characterize other existing algorithms.

VI. Conclusions

We presented a simple window random access algorithm for systems with binary, collision versus noncollision, feedback. We analyzed the algorithm in the presence of the Poisson user model, and under both full feedback sensing and limited feedback sensing. In addition to the throughput and the delay analyses, we studied the effect of feedback errors on the throughput of the algorithm. We also studied the output traffic interdeparture distribution, (The last two studies assume continuous feedback sensing). As compared to the Capetanakis's dynamic algorithm, the proposed algorithm is superior in terms of delays and insensitivity to feedback errors. In contrast to the former, the algorithm can also be easily adapted for implementation under limited feedback sensing, it allows for analytical studying of the output traffic interdeparture distribution, and can be easily analyzed when strict delay limitations are imposed, [3].

VII Appendix

VII.1. Stability Analysis

We will present the stability analysis in the case feedback errors may occur. By setting $\varepsilon = \delta = 0$ in our results, we get the corresponding quantities in the error free case.

The stability region of the algorithm is provided by inequality (1), where the expected value in it, is given by expression (2). We start with the computation of the expected values L_k ; $k \geq 0$ in (2a).

Computation of L_k

We define:

$G_{n,k-n}$: The expected number of slots needed by the algorithm, for the successful transmission of k packets, given that n of those packets have counter values equal to one, and k-n of the packets have counter values equal to two.

Notice that $L_k = G_{k,0}$, for $k \geq 2$, while $L_k \neq G_{k,0}$, for $k < 2$. We first show how to compute L_0 and L_1.

Computation of L_0:
From the operation of the algorithm we have

$$L_0 = \begin{cases} 1 \; ; \; w.p. \; (1-\varepsilon) \\ 1 + G_{0,0} \; ; \; w.p. \; \varepsilon \end{cases} \tag{A1}$$

where

$$G_{0,0} = \begin{cases} 1 + L_0 \; ; \; w.p. \; (1-\varepsilon) \\ 1 + G_{0,0} \; ; \; w.p. \; \varepsilon \end{cases} \tag{A2}$$

From (A1) and (A2) we find that

$$L_0 = \frac{1}{(1-\varepsilon)^2} \tag{A3}$$

Computation of L_1
It was found that L_1 satisfies the following

$$L_1 = \begin{cases} 1 \; ; \; w.p. \; (1-\delta) \\ 1 + G_{1,0} \; ; \; w.p. \; 0.5\delta \\ 1 + G_{0,1} \; ; \; w.p. \; 0.5\delta \end{cases} \tag{A4}$$

where $G_{1,0}$ and $G_{0,1}$ satisfy the following

$$G_{1,0} = \begin{cases} 1+L_0 \; ; \; w.p. \; (1-\delta) \\ 1+G_{1,0} \; ; \; w.p. \; 0.5\delta \\ 1+G_{0,1} \; ; \; w.p. \; 0.5\delta \end{cases} \qquad (A5)$$

$$G_{0,1} = \begin{cases} 1+L_1 \; ; \; w.p. \; (1-\epsilon) \\ 1+G_{0,1} \; ; \; w.p. \; \epsilon \end{cases} \qquad (A6)$$

From (A4), (A5), and (A6) we find that

$$L_1 = \left[1 - \frac{\delta}{2-\delta}\right]^{-1} \left[1 + \frac{\delta}{2(1-\epsilon)} + \frac{\delta}{(2-\delta)}\left[1 + \frac{(1-\delta)}{(1-\epsilon)^2} + \frac{\delta}{2(1-\epsilon)}\right]\right] \qquad (A7)$$

Computation of L_k, for $k\geq 2$.

From the operation of the algorithm we obtain:

$$G_{n,k-n} = 1 + \sum_{i=0}^{n} \binom{n}{i} 2^{-n} G_{i,k-i} \; ; \; n\geq 2 \, , \; k\geq n \qquad (A8)$$

where

$$G_{0,k} = \frac{1}{(1-\epsilon)} + L_k \qquad (A9)$$

$$G_{1,k} = \frac{1 + L_k(1-\delta) + 0.5\delta\left[\dfrac{1}{(1-\epsilon)} + L_{k+1}\right]}{1 - 0.5\delta} \qquad (A10)$$

It can be shown by induction that $G_{n,k-n}$ has the following form

$$G_{n,k-n} = A_n^{(1)} G_{k,0} + A_n^{(2)} G_{k-1,0} + A_n^{(3)} \; ; \; 2\leq n\leq k \qquad (A11)$$

; where $A_n^{(i)}$, i=1,2,3 are independent of k and can be computed recursively as follows:

$$A_2^{(1)} = \frac{2+\delta}{6-3\delta} \, , \; A_2^{(2)} = \frac{4(1-\delta)}{6-3\delta} \, , \; A_2^{(3)} = \frac{14-3\delta-12\epsilon+4\epsilon\delta}{3(1-\epsilon)(2-\delta)} \qquad (A12)$$

$$A_n^{(1)} = [1-2^{-n}]^{-1} \, 2^{-n} \left\{1 + \frac{\delta n}{(2-\delta)} + \sum_{i=2}^{n-1} \binom{n}{i} A_i^{(1)}\right\} \, , \; n\geq 3 \qquad (A13)$$

$$A_n^{(2)} = [1-2^{-n}]^{-1} \, 2^{-n} \left\{\frac{n(1-\delta)}{(1-0.5\delta)} + \sum_{i=2}^{n-1} \binom{n}{i} A_i^{(2)}\right\} \, , \; n\geq 3 \qquad (A14)$$

$$A_n^{(3)} = [1-2^{-n}]^{-1} \left\{1 + \frac{n2^{-n}}{(1-0.5\delta)} + \frac{2^{-n}}{(1-\epsilon)} + \frac{\delta n2^{-n}}{(2-\delta)(1-\epsilon)} + 2^{-n} \sum_{i=2}^{n-1} \binom{n}{i} A_i^{(3)}\right\}, \; n\geq 3 \qquad (A15)$$

For n=k, expression (A11) gives:

$$L_k = G_{k,0} = \frac{A_k^{(2)}}{1-A_k^{(1)}} L_{k-1} + \frac{A_k^{(3)}}{1-A_k^{(1)}} \, , \; k\geq 2 \qquad (A16)$$

Expression (A16) together with the recursions in (A13)-(A15) provide a mean for the

13

computation of L_k, $k \geq 2$.

Development of an upper bound on L_k

It can be seen by induction that:

$$A_k^{(1)} \leq \frac{1}{3} + \frac{\delta}{2-\delta} \; ; k \geq 2 \tag{A17}$$

$$\frac{A_k^{(2)}}{1-A_k^{(1)}} = 1 \; ; k \geq 2 \tag{A18}$$

$$A_k^{(3)} \leq \frac{(2-2\epsilon+\delta)}{(2-\delta)(1-\epsilon)} k + \frac{1}{(1-\epsilon)} \tag{A19}$$

From (A16)-(A19) we finally find:

$$L_k \leq L_{k-1} + \frac{3(2-2\epsilon+\delta)}{(1-\epsilon)(4-5\delta)} k + \frac{3(2-\delta)}{(1-\epsilon)(4-5\delta)} \; , k \geq 2 \tag{A20}$$

from which we can show that

$$L_k \leq \frac{3(2-2\epsilon+\delta)}{2(1-\epsilon)(4-5\delta)} k^2 + \frac{3(6-2\epsilon-\delta)}{2(1-\epsilon)(4-5\delta)} k + \left[L_1 - \frac{6(2-\epsilon)}{(1-\epsilon)(4-5\delta)} \right] \overset{\Delta}{=} L_k^u = \alpha k^2 + \beta k + \gamma \tag{A21}$$

For $\epsilon = \delta = 0$, we obtain the quadratic upper bound. Due to the upper bound on L_k, we conclude then that the following condition is sufficient for stability:

$$\sum_{k=0}^{30} L_k p(k \mid \Delta) + \sum_{k=31}^{\infty} L_k^u p(k \mid \Delta) < \Delta \tag{A22}$$

where

$$p(k \mid \Delta) = e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!}$$

After some manipulations, we conclude that (A22) is equivalent to:

$$f(\lambda \Delta) = \sum_{k=0}^{30} L_k p(k \mid \Delta) + \alpha \left\{ (\lambda \Delta)^2 + \lambda \Delta - \sum_{k=0}^{30} k^2 p(k \mid \Delta) \right\} +$$

$$+ \beta \left[\lambda \Delta - \sum_{k=0}^{30} k \; p(k \mid \Delta) \right] + \gamma \left[1 - \sum_{k=0}^{30} p(k \mid \Delta) \right] < \Delta \tag{A23}$$

Let us now define:

$$x \overset{\Delta}{=} \lambda \Delta \tag{A24}$$

Then, from (A23)-(A24) we conclude that, for the stability of the algorithm, it is sufficient that the input rate λ satisfies the following inequality.

$$\lambda < \sup_{x \geq 0} \frac{x}{f(x)} \tag{A25}$$

The following condition specifies a region of λ values for which the algorithm is unstable.

$$\lambda > \sup_{x \geq 0} \frac{x}{g(x)} \tag{A26}$$

where $g(x) = \sum_{k=0}^{30} L_k e^{-x} x^k / k!$

The maximization of expessions in (A25)-(A26) has been done numerically, and provides the throughput, as well as the optimal window size Δ^*. In all the cases the order of the difference between the two suprema in (A25)-(A26) is less than 10^{-3}. The optimal window size is found as $x^* (\lambda^*)^{-1}$, where x^* is the value that attains the suprema in (A25).

Computation of L_k', for the Capetanakis's Dynamic Algorithm

Here, the quantitites L_k', $k \geq 2$ can be computed recursively as follows:

$$L_k' = [1-2 \, 2^{-k}]^{-1} \left\{ 1 + 2 L_0' \, 2^{-k} + 2 \sum_{i=1}^{k-1} L_i' \binom{k}{i} 2^{-k} \right\} , \quad k \geq 2 \tag{A27}$$

where

$$L_0' = \frac{1}{(1-2\varepsilon)} \text{ and } L_1' = \frac{1-2\varepsilon + \delta}{(1-2\varepsilon)(1-\delta)} \tag{A28}$$

Moreover the following upper bound on L_k', $k \geq 4$, has been found.

$$L_k' \leq \left[\frac{3(1-\varepsilon)}{(1-2\varepsilon)} + \frac{2(\delta-\varepsilon)}{(1-2\varepsilon)(1-\delta)} \right] k - 1 \tag{A29}$$

Using the bounds in (A29), we computed upper and lower bounds on the throughput for the Capetanakis's dynamic algorithm. The computed upper and lower bounds were identical to each other up to the fourth decimal point, and are included in Table 3.

VII. Delay Analysis

The following definitions will be used in the sequel.

l : Length of a conflict resolution interval

τ : Window size.

$E(X/\tau)$: Expected value of the random variable X, given that the window size is τ.

$p(x/\tau)$: The probability that the conflict resolution interval has length x given that the window size is τ.

g_d : Number of slots needed to reach a CRP with lag 1 given that the current lag is equal to d, $d \varepsilon F$.

15

w_d : Cumulative delay experienced by all the packets that were successfully transmitted during g_d slots.

N : Number of packets transmitted in the CRI, that starts at time t.

z : Cumulative delay of the N packets, after the CRP t.

ψ : Cumulative delay of the N packets, until the instant t_2.

Let us also define,

$$G_d = E(g_d) \quad W_d = E(w_d)$$

Note that by definition,

$$W_1 = E(\sum_{n=1}^{Q_1} D_n) = W$$

Also, the mean session length $G = E(T_{i+1} - T_i)$; $i \geq 1$, equals to G_1. If $G_1 < \infty$, then by Wald's identity we have that,

$$Q = \lambda G_1$$

Therefore, the determination of W_1 and G_1 will permit the computation of the mean packet delay.

The operation of the algorithm yields the following relations for the g_d's, d εF.

$$1 \leq d \leq \Delta \quad g_d = l \quad \text{if } l = 1 \tag{B1}$$

$$1 \leq d \leq \Delta \quad g_d = l + g_l \quad \text{if } l > 1 \tag{B1a}$$

and that

$$\Delta < d \quad g_d = l + g_{d-\Delta+l} \tag{B2}$$

Taking expectations in (B1)-(B2) yields:

$$G_d = E(l \mid d) + \sum_{s \neq 1, s \varepsilon F} G_s p(s/d) \quad \text{if } 1 \leq d \leq \Delta, \ d \varepsilon F \tag{B3}$$

$$G_d = E(l \mid \Delta) + \sum_{s \varepsilon F} G_{d-\Delta+s} p(s/\Delta) \quad \text{if } d > \Delta, \ d \varepsilon F \tag{B4}$$

Equations (B3) and (B4) comprise a denumerable system of linear equations. Of interest to us is the element G_1 of a particular solution of this system. We now proceed in the development of an initial upper bound on the solution of the system in (B3)-(B4). Following the methodology in [4], such a bound will be the sequence $G^0_{d,u} = \gamma_u d + \zeta_u$, if γ_u, ζ_u, can be determined so that the following inequalities are satisfied

$$G^0_{d,u} \geq E(l \mid d) + \sum_{s \neq 1, s \varepsilon F} G^0_{s,u} p(s/d) = G^1_{d,u} \quad 1 \leq d \leq \Delta, \ d \varepsilon F \tag{B5}$$

$$G^0_{d,u} \geq E(l \mid \Delta) + \sum_{s \varepsilon F} G^0_{d-\Delta+s} p(s/\Delta) = G^1_{d,u} \quad \text{if } d > \Delta, \ d \varepsilon F \tag{B6}$$

Substituting G^0_d in the right hand side of inequalities (B5) and (B6), it can be seen that if $d \varepsilon F$,

$$G_{d,u}^1 = G_{d,u}^0 + \gamma_u(E(l/d) - d - (1+\lambda d)e^{-\lambda d}) - \zeta_u(1+\lambda d)e^{-\lambda d} \quad \text{if } 1 \le d \le \Delta \tag{B7}$$

$$G_{d,u}^1 = G_{d,u}^0 + E(l/\Delta) - \gamma_u(\Delta - E(l/\Delta)) \quad \text{if } d > \Delta \tag{B8}$$

From (B8) we conclude that if $E(l/\Delta) < \Delta$, the condition for stability of the system, inequalities (B6) are satisfied if,

$$\gamma_u = \frac{E(l|\Delta)}{\Delta - E(l/\Delta)} \tag{B9}$$

With this value of γ_u, it can be seen that inequalities (B5) are satisfied if

$$\zeta_u = \max\{-\gamma_u, \sup_{1 \le d \le \Delta}(\theta(d))\} \tag{B10}$$

where

$$\theta(d) = \frac{E(l/d) + \gamma_u[E(l|d) - d - (1+\lambda d)e^{-\lambda d}]}{(1+\lambda d)e^{-\lambda d}} \tag{B11}$$

From the above discussion we conclude that the solution to system (B3)-(B4) satisfies the inequalities

$$G_d \le \gamma_u d + \zeta_u, \quad d\varepsilon F \tag{B12}$$

The uniqueness of the solution is guaranteed by the same techniques as in [4]. If we use a similar method for the development of a lower bound, we find that

$$\gamma_l d + \zeta_l = G_d, \quad d\varepsilon F \tag{B13}$$

where

$$\gamma_l = \gamma_u, \quad \text{and} \quad \zeta_l = \inf_{1 \le d \le \Delta}(\theta(d)) \tag{B14}$$

From the operation of the algorithm we also have that W_d ; $d\varepsilon F$ satisfy the following system of linear equations,

$$W_d = E(z \mid d) + E(\psi \mid d) + \sum_{s\varepsilon F, s \ne 1} W_s\, p(s \mid d), \quad 1 \le d \le \Delta, \ d\varepsilon F \tag{B15}$$

$$W_d = E(z \mid \Delta) + E(\psi \mid \Delta) + \sum_{s\varepsilon F} W_{d-\Delta+s}\, p(s \mid \Delta), \quad \Delta < d, \ d\varepsilon F \tag{B16}$$

Following the methodology in [4] we can show that

$$W_{d,l}^0 = \mu_l d^2 + \nu_l d + \xi_l \le W_d \le \mu_u d^2 + \nu_u d + \xi_u = W_{d,u}^0 \tag{B17}$$

where,

$$\mu_u = \mu_l = \frac{\lambda\Delta}{2(\Delta - E(l \mid \Delta))} \tag{B18}$$

$$v_u = v_l = \frac{E(z \mid \Delta) + E(\psi \mid \Delta) - \lambda \Delta^2 + \mu_u E\{(\Delta - l)^2 \mid \Delta\}}{\Delta - E(l \mid \Delta)} \tag{B19}$$

$$\xi_u = \sup_{1 \le d \le \Delta} (\phi(d)), \quad \xi_l = \inf_{1 \le d \le \Delta} (\phi(d)) \tag{B20}$$

where,

$$\phi(d) = \frac{E(z \mid d) + E(\psi \mid d) + \mu_u(E(l^2 \mid d) - d^2 - (1 + \lambda d)e^{-\lambda d}) - v_u(d - E(l \mid d) - (1 + \lambda d)e^{-\lambda d})}{(1 + \lambda d)e^{-\lambda d}} \tag{B21}$$

Substituting $W_{d,.}^0$ in the right hand side of (B15)-(B16), we obtain $W_{d,.}^0$. We find,

$$W_{1,u}^1 = E(z \mid 1) + E(\psi \mid 1) + \mu_u(E(l^2 \mid 1) - (1 + \lambda)e^{-\lambda}) + v_u(E(l \mid 1) - (1 + \lambda)e^{-\lambda}) + \xi_u(1 - (1 + \lambda)e^{-\lambda}) \tag{B22}$$

$$W_{1,l}^1 = W_{1,u}^1 - (\xi_u - \xi_l)(1 - (1 + \lambda)e^{-\lambda}) \tag{B23}$$

Also,

$$G_{1,u}^1 = E(l \mid 1) + \gamma_u(E(l \mid 1) - (1 + \lambda)e^{-\lambda}) + \zeta_u(1 - (1 + \lambda)e^{-\lambda}) \tag{B24}$$

$$G_{1,l}^1 = G_{1,u}^1 - (\zeta_u - \zeta_l)(1 - (1 + \lambda)e^{-\lambda}) \tag{B25}$$

From the regenerative theorem [4], we have that

$$\mathbf{D}^l = \frac{W_{1,l}^1}{\lambda G_{1,u}^1} \le \mathbf{D} \le \frac{W_{1,u}^1}{\lambda G_{1,l}^1} = \mathbf{D}^u \tag{B26}$$

In this Appendix we also show that the conditional expectations of the form $E(X \mid d)$ can be computed with high accuracy. Let us define
$E(X \mid d,k)$: The conditional expectation of the random variable X, given that the arrival interval contains k packets, and has length d. Then,

$$E(X \mid d) = \sum_{k=0}^{\infty} E(X \mid d,k) e^{-\lambda d} \frac{\lambda d^k}{k!} \tag{B27}$$

The quantities $E(X \mid d,k)$ depend only on k. In Section V.I we show that the quantities $L_k = E(l \mid d,k)$ can be computed recursively:

$$L_0 = L_1 = 1 \tag{B28}$$

$$L_k = \frac{A_k^{(2)}}{1 - A_k^{(1)}} L_{k-1} + \frac{A_k^{(3)}}{1 - A_k^{(1)}} \; ; k \ge 2 \tag{B29}$$

where $A_n^{(i)}$, $i = 1,2,3$ can be computed recursively as follows:

$$A_n^{(1)} = [1 - 2^{-n}]^{-1} \, 2^{-n} \left\{ 1 + \sum_{i=2}^{n-1} A_i^{(1)} \binom{n}{i} \right\}, \quad n \ge 3 \tag{B30}$$

18

$$A_n^{(2)} = [1 - 2^{-n}]^{-1} 2^{-n} \left\{ n + \sum_{i=2}^{n-1} A_i^{(2)} \binom{n}{i} \right\}, \quad n \geq 3 \tag{B31}$$

$$A_n^{(3)} = [1 - 2^{-n}]^{-1} \left\{ 1 + 2^{-n}(n+1) + 2^{-n} \sum_{(i=2)}^{n-1} A_i^{(3)} \binom{n}{i} \right\}, \quad n \geq 3 \tag{B32}$$

$$A_2^{(1)} = 1/3, \ A_2^{(2)} = 2/3, \ A_2^{(3)} = 7/3 \tag{B33}$$

The quantities $Z_k = E(z \mid d, k)$, can be also computed recursively:

$$Z_0 = 0, \quad Z_1 = 1 \tag{B34}$$

$$Z_k = \frac{A_k^{(5)}}{1 - A_k^{(4)}} Z_{k-1} + \frac{A_k^{(6)}}{1 - A_k^{(4)}} \ ; \ k \geq 2 \tag{B35}$$

and the quantities $A_k^{(i)}$, $i = 4, 5, 6$ can be computed recursively as follows:
$A_n^{(4)}$; $n \geq 3$ satisfies the recursion in (B30)
$A_n^{(5)}$; $n \geq 3$ satisfies the recursion in (B31)

$$A_n^{(6)} = [1 - 2^{-n}]^{-1} \{1 + n2^{-n} + n^2 2^{-n} + 2^{-n} \sum_{i=2}^{n-1} \binom{n}{i} A_i^{(6)} \} \ , \ n \geq 3 \tag{B36}$$

$$A_2^{(4)} = \frac{1}{2}, \ A_2^{(5)} = \frac{1}{2}, \ A_2^{(6)} = \frac{7}{2} \tag{B37}$$

The quantities $Y_k = E(l^2 \mid d, k)$ can be computed as follows:

$$Y_0 = 0, \quad Y_1 = 1 \tag{B38}$$

$$Y_k = \frac{A_k^{(8)}}{1 - A_k^{(7)}} Y_{k-1} + \frac{A_k^{(9)}}{1 - A_k^{(7)}} \ ; \ k \geq 2 \tag{B39}$$

and the quantities $A_k^{(i)}$, $i = 7, 8, 9$ can be computed recursively as follows:
$A_n^{(7)}$; $n \geq 3$ satisfies the recursion (B31)
$A_n^{(8)}$; $n \geq 3$ satisfies the recursion (B32)

$$A_n^{(9)} = [1 - 2^{-n}]^{-1} \{2L_n - 1 + 2^{-n}(1 + 2L_n) + n2^{-n}(1 + L_{n-1}) + 2^{-n} \sum_{i=2}^{n-1} \binom{n}{i} A_i^{(9)} \} \ n \geq 3 \tag{B40}$$

$$A_2^{(7)} = \frac{1}{3}, \ A_2^{(8)} = \frac{2}{3}, \ A_2^{(9)} = 16 \tag{B41}$$

From the above formulas, we see that a finite number, M, of terms from the infinite series in (B27), can be computed. Also, for large k values, and based on the recursive expressions, simple upper and lower bounds on $E(X \mid d, k)$ can be developed. Those bounds can be used to tightly bound the sum

$$\sum_{k=M+1}^{\infty} E(X \mid d, k) e^{-\lambda d} \frac{(\lambda d)^k}{k!}$$

Remark It can be also proved that $E(\psi \mid d) = \frac{\lambda d^2}{2}$.

19

V.III Interdeparture Distribution Analysis

We first provide some definitions.

$l_{k,m}$: Given k packets with counter values equal to 1 and m packets with counter values equal to 2, the number of slots needed by the algorithm until the first successful transmission, (and including it), after the k-multiplicity collision has been observed.

$n_{k,s}$: Given a collision resolution interval which starts with a k-multiplicity collision, the number of length s interdeparture intervals within it. The length from the initial collision to the first successful transmission is included in the counting.

h_d: Starting with a CRP at which the lag equals d, $d \geq 1$, and which follows a successful transmission, the number of slots needed by the algorithm to reach the first lag-one CRP which follows a slot containing a successful transmission.

$m_{d,s}$: Starting with a CRP at which the lag equals d, $d \geq 1$, and which follows a successful transmission, the number of length s interdeparture intervals until the first lag-one CRP which follows a slot containing a successful transmission. The distance from the initial CRP to the first successful transmission is included in the counting.

$P(k,l,\delta \mid d)$: Given an arrival interval of length d, the probability that there are k arrivals in it, that $l_{k,0} = \delta$, and that it takes l slots for its resolution, including the initial collision slot.

$P_k(l)$: Given a k-multiplicity initial collision, the probability that it takes l slots for its resolution, including the initial collision slot.

The above definitions are needed for the derivation of recursions that are pertinent to the infinite-dimensionality systems associated with the quantities in (9). We first note that:

$$H = E\{h_1\} \ , \ C = \lambda H$$

$$E\{\sum_{n=1}^{C_1} l_n(s)\} = E\{m_{1,s}\}$$

Auxiliary Recursions

The operations of the algorithm induce the following recursions:

(I)
$$l_{1,m} = 0; \forall m \ , \ P(l_{k,m}=0)=0 \ ; \ \forall k \geq 2, \ \forall m$$

$$l_{0,m} = 1 + l_{m,0} \ , \ P(l_{0,m} = 1) = \begin{cases} 1 \ , \text{if } m=1 \\ 0 \ , \text{if } m \neq 1 \end{cases}$$

$$l_{k,m} = 1 + l_{i,m+k-i} \; ; \; \text{with probability} \begin{pmatrix} k \\ i \end{pmatrix} 2^{-k}, \; k \geq 2$$

$$P(l_{k,m} = s) = \begin{cases} 1 \; , \; \text{if } k=1 \text{ and } s=0 \\ P(l_{m,0} = s-1) \; , \; \text{if } k=0, \; s \geq 1 \\ 2^{-k} \sum_{i=0}^{k} \begin{pmatrix} k \\ i \end{pmatrix} P(l_{i,\,m+k-i} = s-1), \; \text{if } k \geq 2, \; s \geq 1 \end{cases}$$

(II)

$$n_{1,s} = \begin{cases} 1 \; , \; \text{if } s=1 \\ 0 \; , \; \text{if } s \neq 1 \end{cases}$$

$$k \geq 2 \; ; \; n_{k,s} = \begin{cases} n_{k-1,s} \; , \; \text{with probability } P(l_{k,0} \neq s-1) \\ 1 + n_{k-1,s} \; , \; \text{with probability } P(l_{k,0} = s-1) \end{cases}$$

$$N_{k,s} \overset{\Delta}{=} E\{n_{k,s}\} = \begin{cases} \sum_{i=2}^{k} P(l_{i,0} = s-1) \; , \; \text{if } s > 1 \\ 1 + \sum_{i=2}^{k} P(l_{i,0} = 0) = 1, \; \text{if } s = 1 \end{cases}$$

(III)

$$k=0,1 \; ; \; P_k(l) = \begin{cases} 1 \; , \; \text{if } l = 1 \\ 0 \; , \; \text{otherwise} \end{cases}, \; P_2(l) = P(l_{2,0} = l-2), \; \text{for } l \geq 3$$

$$\left. \begin{array}{c} k \geq 2 \\ l \geq k+2 \end{array} \right\} \; ; \; P_k(l) = \sum_{s=1}^{l-k-1} P(l_{k,0} = s) \, P_{k-1} \, (l-s-1)$$

Given Poisson intensity λ,

$$P(k,l,\rho \mid d) = e^{-\lambda d} \, \frac{(\lambda d)^k}{k!} \, P(l_{k,0} = \rho) P_{k-1} \, (l-\rho-1)$$

Recursions for h_d

21

Given Poisson intensity λ, and from the operations of the algorithm, we easily conclude:

$$d \leq \Delta \; ; \; h_d = \begin{cases} 1 \; , \text{ with probability } \lambda d e^{-\lambda d} \\[2mm] 1 + h_1 \; , \text{ with probability } e^{-\lambda d} \\[2mm] l + h_l \; , \text{ with probability } \sum_{k=2}^{\infty} e^{-\lambda d} \dfrac{(\lambda d)^k}{k!} P_k(l), \; l \geq 2 \end{cases}$$

$$d > \Delta \; ; \; h_d = l + h_{d-\Delta+l} \; , \text{ with probability } \sum_{k=0}^{\infty} e^{-\lambda \Delta} \dfrac{(\lambda \Delta)^k}{k!} P_k(l)$$

and thus,

$$H_d \stackrel{\Delta}{=} E\{h_d\} = e^{-\lambda d} + E\{l \mid d\} + e^{-\lambda d} H_1 + \sum_{k \geq 2} \sum_{l \geq 2} e^{-\lambda d} \frac{(\lambda d)^k}{k!} P_k(l) H_l \; ; \quad d \leq \Delta$$

$$H_d = E\{l \mid \Delta\} + \sum_{k \geq 0} \sum_{l \geq 1} e^{-\lambda \Delta} \frac{(\lambda \Delta)^k}{k!} P_k(l) H_{d-\Delta+l} \; ; \quad d > \Delta$$

where,

$$E\{l \mid d\} = \sum_{k \geq 0} \sum_{l \geq 1} e^{-\lambda d} \frac{(\lambda d)^k}{k!} P_k(l).l$$

Recursions for $m_{d,s}$

For $\lfloor \; \rfloor$ denoting integer part , for w.p. meaning with probability, and for Poisson intensiy λ, we conclude:

$$d \leq \Delta; \; m_{d,s} = \begin{cases} n_{1,s} \; ; \; w.p. \; e^{-\lambda d} \lambda d \\[2mm] n_{k,s} + m_{l,s} \; ; \; w.p. \; e^{-\lambda d} \dfrac{(\lambda d)^k}{k!} P_k(l) \; ; \; k \geq 2 \\[3mm] n_{k-1,s} + m_{l,s} \; ; \; w.p. \; P(0 \mid d) \underset{\substack{n \geq 0 \\ n+\rho+2 \neq s}}{\Sigma} P^n(0 \mid 1) P(k,l,\rho \mid 1) \; ; \; k \geq 2 \\[4mm] 1 + n_{k-1,s} + m_{l,s} \; ; \; w.p. \; e^{-\lambda(d+1)} \dfrac{\lambda^k}{k!} \underset{\substack{n \geq 0 \\ n+\rho+2 = s}}{\Sigma} e^{-\lambda n} P(l_{k,0}=\rho) P_{k-1}(l-\rho-1); k \geq 2 \\[3mm] 1 \; ; \; w.p. \; \lambda e^{-\lambda(d+s-1)} \; ; \; s \geq 2 \end{cases}$$

For $d > \Delta$:

22

$$m_{d,s} = n_{k,s} + m_{d-\Delta+l,s} \quad ; w.p. \, e^{-\lambda\Delta} \frac{(\lambda\Delta)}{k!^k} P_k(l) \; ; \; k \geq 1$$

$$= n_{k-1,s} + m_{d-n(\Delta-1)+l,s} \quad ; w.p. \sum_{\substack{\rho \neq s-n-1 \\ 1 \leq n \leq \lfloor \frac{d-\Delta}{\Delta-1} \rfloor}} e^{-\lambda\Delta(n+1)} \frac{(\lambda\Delta)^k}{k!} P(l_{k,0}=\rho) \, P_{k-1}(l-\rho-1), \; k \geq 1, \, \text{if} \, \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \geq 1$$

$$= 1 + n_{k-1,s} + m_{d-n(\Delta-1)+l,s} \quad ; w.p. \sum_{1 \leq n \leq \lfloor \frac{d-\Delta}{\Delta-1} \rfloor} e^{-\lambda\Delta(n+1)} \frac{(\lambda\Delta)^k}{k!} P(l_{k,0}=s-n-1) \, P_{k-1}(l-s+n), \; k \geq 1,$$

$$; \text{if} \, \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \geq 1$$

$$n_{k-1,s} + m_{l,s} \; ; \; w.p. \; e^{-\lambda \left[d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right]} \frac{\left[\lambda(d - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor (\Delta-1)) \right]^k}{k!} \sum_{\lfloor \frac{d-\Delta}{\Delta-1} \rfloor + \rho + 1 \neq s} P(l_{k,0}=\rho) P_{k-1}(l-\rho-1) \; ; \; k \geq 2$$

$$= 1 + n_{k-1,s} + m_{l,s} \; ; \; w.p. \; e^{-\lambda \left[d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right]} \frac{\left[\lambda \left[d - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor (\Delta-1) \right] \right]^k}{k!} P \left[l_{k,0} = s - 1 - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right] .$$

$$. P_{k-1} \left[l - s + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right] \; ; \; k \geq 2$$

$$n_{k-1,s} + m_{l,s} \; ; \; w.p. \, e^{-\lambda(d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor + 1)} \sum_{\substack{n \geq 0 \\ n + 2 + \rho + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \neq s}} e^{-\lambda n} \frac{\lambda^k}{k!} P(l_{k,0}=\rho) P_{k-1}(l-\rho-1) \; ; \; k \geq 2$$

$$= 1 + n_{k-1,s} + m_{l,s} \; ; \; w.p. \, e^{-\lambda \left[d + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor + 1 \right]} \sum_{n \geq 0} e^{-\lambda n} \frac{\lambda^k}{k!} P \left[l_{k,0} = s - n - 2 - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \right] .$$

$$. P_{k-1} \left[l + n - s + \lfloor \frac{d-\Delta}{\Delta-1} \rfloor + 1 \right] \; ; \; k \geq 2$$

$$= 1 \; ; \; w.p. \, \lambda e^{-\lambda(d+s-1)} \; ; \, \text{if} \, s - 2 - \lfloor \frac{d-\Delta}{\Delta-1} \rfloor \geq 0$$

$$= 1 \; ; \; w.p. \lambda \left[d - \left\lfloor \frac{d-\Delta}{\Delta-1} \right\rfloor (\Delta-1) \right] e^{-\lambda \left[d + \left\lfloor \frac{d-\Delta}{\Delta-1} \right\rfloor \right]} \; ; \; \text{if } s = 1 + \left\lfloor \frac{d-\Delta}{\Delta-1} \right\rfloor$$

Let us define,

$$U(x) \overset{\Delta}{=} \begin{cases} 1 \; , \; x \geq 0 \\ 0 \; , \; x < 0 \end{cases}$$

$$P_\delta(l) \overset{\Delta}{=} \sum_{k \geq 1} e^{-\delta} \frac{\delta^k}{k!} P_k(l)$$

$$N_{\delta,s} \overset{\Delta}{=} \sum_{k \geq 1} N_{k,s} \, e^{-\delta} \frac{\delta^k}{k!} \; , \; N_{\delta,1} = 1 - e^{-\delta}$$

$$P_{\delta,\rho} \overset{\Delta}{=} \sum_{k \geq 1} e^{-\delta} \frac{\delta^k}{k!} P(l_{k,0} = \delta)$$

Then, using the above defined quantities, and the recursions on $m_{d,s}$, we easily find:

For $d \leq \Delta$:

$$M_{d,s} \overset{\Delta}{=} E\{m_{d,s}\} = N_{\lambda d,s} + \sum_{l \geq 3} M_{l,s} \left[P_{\lambda d}(l) + \frac{e^{-\lambda d}}{1-e^{-\lambda}} P_\lambda(l) \right] +$$

$$+ \frac{e^{-\lambda d}}{1-e^{-\lambda}} \left[N_{\lambda,s} - P_{\lambda,s-1} \right] + U(s-2)e^{-\lambda d} \left\{ e^{-\lambda(s-2)} \sum_{m=0}^{s-2} e^{\lambda m} P_{\lambda,m} + \right.$$

$$+ \lambda e^{-\lambda(s-1)} - \lambda \frac{e^{-\lambda}}{1-e^{-\lambda}} \right\}$$

For $d > \Delta$ and $p \overset{\Delta}{=} \left\lfloor \frac{d-\Delta}{\Delta-1} \right\rfloor$, $p = 0,1,...$:

$$M_{d,s} = N_{\lambda\Delta,s} + \sum_{l \geq 1} M_{d-\Delta+l,s} P_{\lambda\Delta}(l)$$

$$+ e^{-\lambda\Delta p} \left\{ N_{\lambda(d+p-p\Delta),s} - P_{\lambda(d+p-p\Delta),s-1} + \sum_{l \geq 3} M_{l,s} P_{\lambda(d+p-p\Delta)}(l) \right\}$$

$$+ \frac{e^{-\lambda(d+p)}}{1-e^{-\lambda}} \left\{ N_{\lambda,s} - P_{\lambda,s-1} + \sum_{l \geq 3} M_{l,s} P_\lambda(l) \right\}$$

24

$$+ U(p-1)\left\{ \frac{e^{-\lambda\Delta}(1-e^{-\lambda\Delta p})}{1-e^{-\lambda\Delta}} \left[N_{\lambda\Delta,s} - P_{\lambda\Delta,s-1} \right] + \sum_{1\leq n\leq p} e^{-\lambda\Delta n} \sum_{l\geq 1} M_{d-n(\Delta-1)+l,s}\, P_{\lambda\Delta}(l) \right\}$$

$$+ U(p-1)U(s-2)e^{-\lambda(s-1)} \sum_{m=s-1-min(p,\, s-1)}^{s-2} e^{\lambda m}\, P_{\lambda\Delta,\, m}$$

$$- U(s-2)\lambda e^{-\lambda(d+p)} \left[(d+p-p\Delta)e^{\lambda p\Delta} + \frac{e^{-\lambda}}{1-e^{-\lambda}} \right]$$

$$+ U(s-1-p)\, e^{-\lambda\Delta p}\, P_{\lambda(d+p-p\Delta),s-1-p}$$

$$+ U(s-2-p)\, e^{-\lambda(d+s-2)} \sum_{m=0}^{s-2-p} e^{\lambda m}\, P_{\lambda,m}$$

Bounds

For the numbers $N_{k,s}$, we used the following bounds:

$$0 \leq N_{k,s} \leq k-1 \; ; \; \forall s$$

Regarding the numbers H_d, we used the methodology in [4], and proved that,

$$\alpha_l d + \beta_l \leq H_d \leq \alpha_u\, d + \beta_u \; , \; d \geq 1$$

where,

$$\alpha_l = \alpha_u = [\Delta - E\{l\,|\,\Delta\}]^{-1}\, E\{l\,|\,\Delta\}$$

$$\beta_l = \inf_{1\leq d\leq\Delta} Q(d) \; , \; \beta_u = \max\left[-\alpha_u \, , \, \sup_{1\leq d\leq\Delta} Q(d) \right]$$

for:

$$Q(d) = \left[\lambda d e^{-\lambda d} \right]^{-1} \left\{ E\{l\,|\,d\} + \alpha_u\left[E\{l\,|\,d\} - d - \lambda d e^{-\lambda d} \right] \right\}$$

Bounds on the numbers $M_{d,s}$ can be developed similarly with those for the numbers H_d. The former are significantly more complicated, however. Instead, we used the following simpler and intuitively clear bounds, where H_d^u denotes the upper bound on the quantity H_d:

$$0 \leq M_{d,s} \leq H_d^u$$

We used the above bounds, for $d \geq 30$.

U: unresolved interval   ct: current time
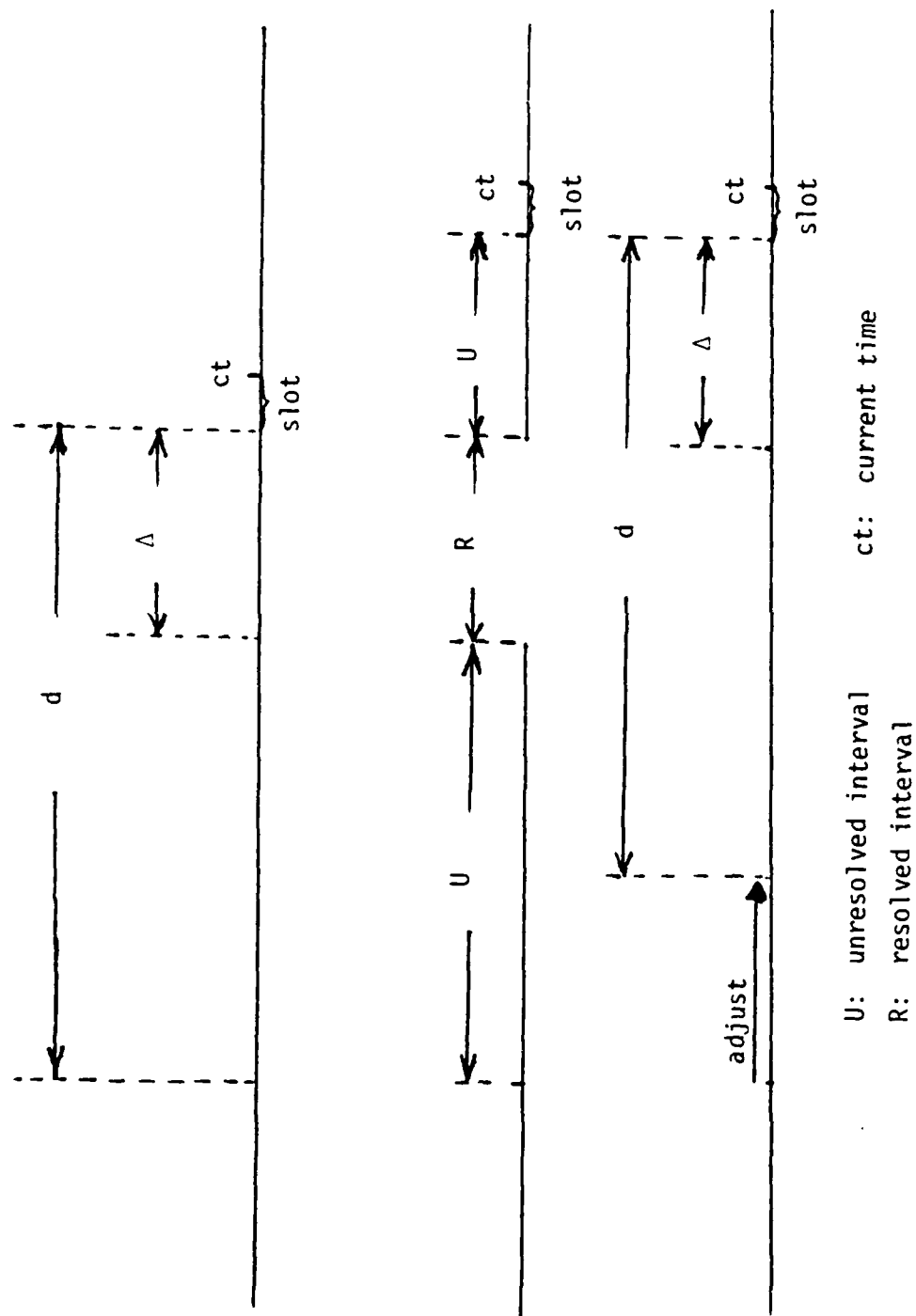R: resolved interval

Figure 1

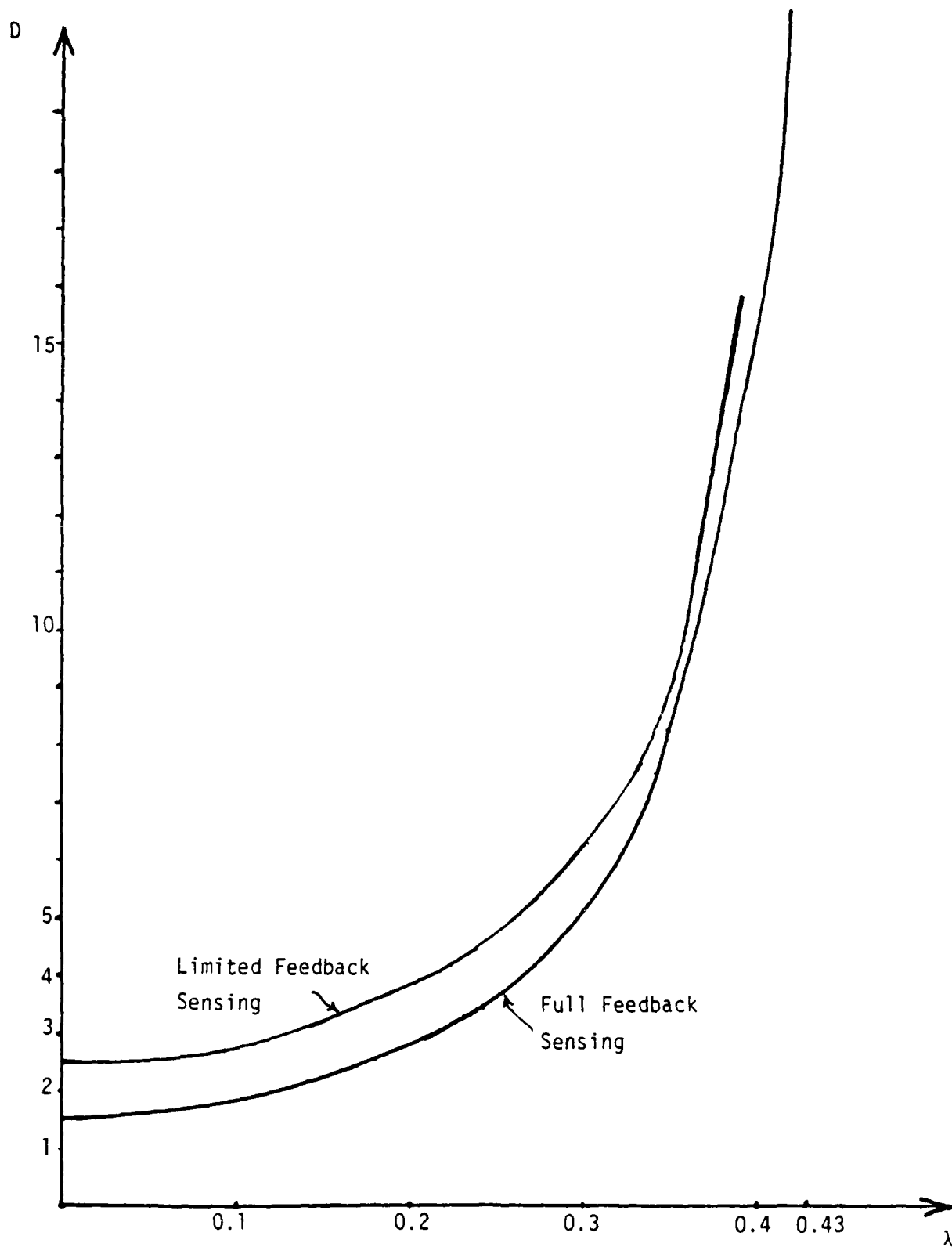Window Selection in the Limited Sensing Environment

Figure 2

Expected Delays Induced by the Proposed Algorithm
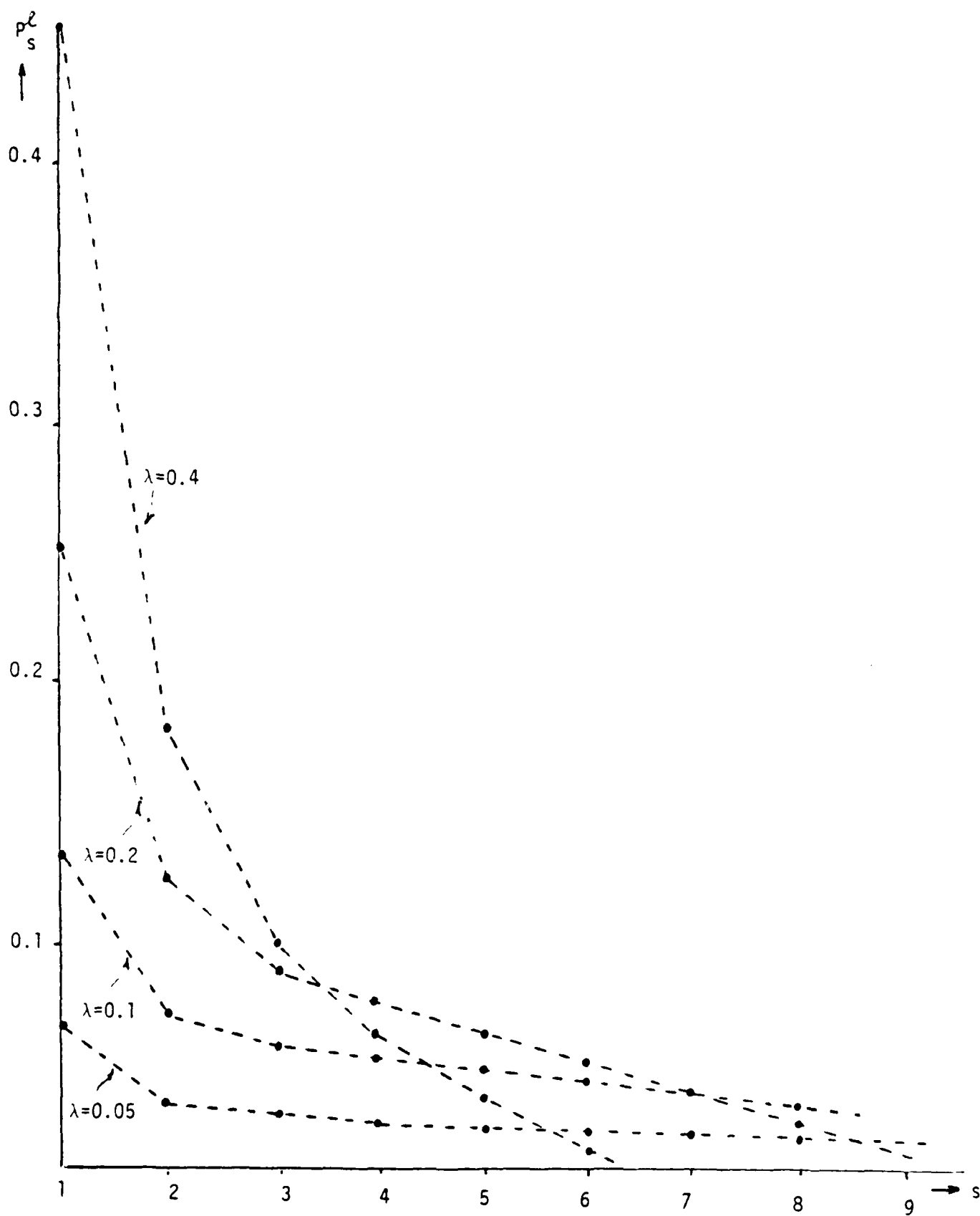in the Full Feedback Sensing and the Limited Feedback
Sensing Environments

Figure 3

Lower Bounds of the Interdeparture Distribution
for Various Poisson Rates

References

[1] J. I. Capetanakis, "Tree Algorithms for Packet Broadcast Channel, IEEE Trans. Inform. Theory, vol. IT-25, pp. 505-515, Sept. 1979.

[2] R. G. Gallager, "Conflict Resolution in Random Access Broadcast Networks", in Proc. AFOSR Workshop Communication Theory and Applications, Provincetown, MA, Sept. 1978, pp. 74-76.

[3] M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, "A Full Sensing Window Random-Access Algorithm for Messages with Strict Delay Constraints", Report No. UVA/525415/EE87/103, School of Engineering and Applied Science, University of Virginia, Charlottesville, Virginia, Feb. 1987. Also, Algorithmica, An International Journal in Computer Science, Springer - Verlag, to appear in 1988.

[4] L. Georgiadis, L. Merakos, and P. Papantoni-Kazakos, "A Method for the Delay Analysis of Random Multiple Access Algorithms whose Delay Process is Regenerative", IEEE Journal on Selected Areas in Communications, vol. SAC-5, pp. 1051-1062, July 1987.

[5] M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, "On the Relation between the Finite and the Infinite Population Models for a class of RAAs", IEEE Transactions on Communications, Vol. COM-35, Nov. 1987, pp.1239-1240.

[6] S. Stidham, Jr., "Regenerative Processes in the Theory of Queues with applications to the Alternating-Priority Queue", Adv. Appl. Prob., Vol. 4, pp. 542-577, 1972.

[7] J. L. Massey, "Collision Resolution Algorithms and Random Access Communications", in Multi-User Communication, Ed. G. Longo, Springer-Verlag, CISM.

[8] P. Studer and H. Pletscher, "Q-ary Part-and-Try Algorithm for Packet Conflict Resolution", Inst. Telecommun. Swiss Fed. Inst. Technol., Zurich, Switzerland, Oct. 11, 1984.

[9] L. Georgiadis and P. Papantoni-Kazakos, "A 0.487 Throughput Limited Sensing Algorithm", IEEE Trans. on Information Theory, Vol. IT-33, No. 2, March 1987, pp. 233-237.

[10] P. Humblet, "On the Throughput of Channel Access Algorithms with Limited Sensing", IEEE Trans. on Communications, Vol. COM-34, April 1986, pp 345-347.

[11] A. G. Pakes, "Some Conditions for Ergodicity and Recurrence of Markov Chains", Oper. Res. Vol. 21, pp. 617-622, 1973.

[12] M. Kaplan, "A Sufficient Condition for Nonergodicity of a Markov Chain", IEEE Trans. Info. Theory, Vol. IT-25, pp. 470-471, 1979.

DISTRIBUTION LIST

15	Dr. G. Malecki Office of Naval Research Code 1142 800 N. Quincy Street Arlington, VA 22217-5000
16	Dr. David Kleinman ESE Department, U-157 The University of Connecticut STORRS, CT 06268
17 - 28	Defense Technical Information Center, S47031 Building 5, Cameron Station Alexandria, VA 22314
29 - 30	P. Kazakos, EE
31	M. Paterakis
32 - 33	E. H. Pancake, Clark Hall
34	SEAS Publications Files

*	Office of Naval Research Resident
	   Representative
	818 Connecticut Avenue, N. W.
	Eighth Floor
	Washington, D.C. 20006
	
	Attention: Mr. Michael McCracken
	            Administrative Contracting
	               Officer

*Send Cover Letter Only

JO#1098:ph

END

DATE
FILMED
6-88
DTIC